Publishing JD Edwards Real Time Events to Oracles Enterprise Service Bus
Overview

In this tutorial you will be publishing JD Edwards EnterpriseOne Real Time Events (RTE) to Oracles Enterprise Service Bus (ESB). This process utilizes JMS Queues as the mechanism for passing the Real Time Events from JD Edwards EnterpriseOne to ESB.

This Tutorial leverages the Reference Implementation (RI) for JD Edwards EnterpriseOne Real Time Events to ESB (RTForOutbound.zip) – which can be found on the “system\classes\samples\ESB\” directory of the JD Edwards EnterpriseOne install.

There are several steps to configure, test, and run this process – including:

1. Creating the JMS Queue Connection Factory.
2. Configuring the JD Edwards Transaction Server to publish RTE’s to JMS Queue.
3. Test publishing RTE from JD Edwards EnterpriseOne to JMS Queue.
4. Import the Reference Implementation to JDev.
5. Register and Test the sample on the ESB Server.

Pre-reqs:

- You are familiar with JD Edwards EnterpriseOne Server Manager
- You have installed SOA suite with ESB console
- You have installed JDeveloper 10.1.3.1.0
- You have found RTForOutbound.zip in the system\classes\samples\ESB\ directory, and you have extracted the file.

For your note: Write down all your machine information.

<table>
<thead>
<tr>
<th>Server/Machine</th>
<th>Machine Information</th>
<th>Example:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enterprise Server:</td>
<td></td>
<td><a href="http://jde_enterpriseone:5800">http://jde Enterpriseone:5800</a></td>
</tr>
<tr>
<td>Server Manager:</td>
<td></td>
<td>[<a href="http://jde">http://jde</a> Manage:8999/manage](<a href="http://jde">http://jde</a> Manage:8999/manage) jde admin/jde_admin</td>
</tr>
</tbody>
</table>
Step 1: Create JMS Queue and Connection Factory in OAS (where Transaction Server is installed)

1. Login to admin console of OAS where the Transaction Server is installed.
   note: we installed Transaction Server in CONTAINER3 instance

2. Click on the CONTAINER3 link and click on the Administration link.
3. Click on the “Go to Task” icon for “JMS Connection Factories”

4. Create New and enter the values as follows:

- ConnectionFactoryType = Queue
- JNDI Location = jms/ESBSubscriberQCF
- XA Enabled – Select
5. Go back to Administration window like step 2 above and click on “Got to Task” for “JMS Destinations”.

6. Click Create New and enter values as follow:

   Destination Type = Queue
7. Newly created Queue will be appeared like the screenshot in step number 5.

8. Edit jas.ini of your Transaction Server from Server Manager and click on the “Real Time Events” under Configuration (see below).

   OAS JNDI User: oc4jadmin (user for OAS/TS)
   OAS JNDI Password: oc4jadmin (Password for OAS/TS)

   Note: you cannot directly change this in your jas.ini you must do it from server manager.
9. Add a new log file for **EVENPROCESSOR** for troubleshooting purpose. – click on "jdelog.properties logging" link under "configuration" in Server Manager. Add info below:

10. Start the transaction server from Server Manager. And monitor your log. The rtexxxxx.log should grow about 32K and it should stop growing. If this keeps increasing then you might be experiencing some errors in the logs. (X:\agent\targets\E1_TransServer_96\logs)

11. **STOP THE Transaction Server from Server Manager.**
Step 2: Create ESB Subscriber Information in JDE E1 (P90702A)

1. Create an E1 user for an ESB subscriber – make sure you can log in to HTML client using the new user ID. (I created a user ID called **ESB**)
2. Go to Fast Path and type **P90702A**
3. Click Add and enter following values:

<table>
<thead>
<tr>
<th>Subscriber:</th>
<th>ESB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transport Type:</td>
<td>JMSQUEUE</td>
</tr>
<tr>
<td>Delivery Format:</td>
<td>XML</td>
</tr>
<tr>
<td>Application Server:</td>
<td>OAS</td>
</tr>
<tr>
<td>Queue location</td>
<td>LOCAL</td>
</tr>
<tr>
<td>QueueConnectionFactoryJNDI</td>
<td>jms/ESBSubscriberQCF</td>
</tr>
<tr>
<td>Queue Name:</td>
<td>jms/ESBSubscriberQueue</td>
</tr>
<tr>
<td>Message Format:</td>
<td>XML</td>
</tr>
<tr>
<td>ContextFactory</td>
<td>com.evermind.server.rmi.RMIInitialContextFactory</td>
</tr>
<tr>
<td>providerURL</td>
<td>opmn:ormi://dencmassive.mlab.jdedwards.com:6003:CONTAINER3</td>
</tr>
</tbody>
</table>
opmn:ortmi://YourTransactionServerName:port:TSInstanceName

port: this is OPMN port number. From your SOA suite machine, type the following command to find out the port number.
SOA install\opmn\bin\opmnctl status –port
Repose should be: DENCCAPITOL:6003

TSInstanceName: The name of the instance of your transaction server. See screenprint of Step 1 #1.

Note: You can update the processing option for P90702A for Contextfactory and provider URL. So, next time you create another subscriber, the correct info gets populated:

4. Add RTABOUT(Event) and JDV812 (environment) and activate it.
5. RESTART E1
Note: If you have any other subscribers, you might want to disable them except ESB subscriber for testing.

Step 3: Test RTE Message from JDE E1 to ESB Subscriber Queue.

1. Create an Address Book info from P01012
2. At this point, you should have Transaction Server stopped (see Step 1 #11), so make sure you see a RTABOUT message in F90710.
3. If you see the message, then start the Transaction Server
   a. Transaction Server will pick up the message from F90710 and the message passes through the ESBQueue00 and it ends up in ESBSubscriberQueue.
   b. Open O:\product\10.1.3.1\OracleAS_1\j2ee\CONTAINER3\persistence\CONTAINER3_default_group_1 folder and see if you have more than 1KB for the ESBSubscriberQueue.
c. You can check to see if you have a message in ESBSubscriberQueue using OAS. Please review the
guide “How to view the queue.doc”. If it is working successfully, then you’ll see a message in
the “ESBSubscriberQueue” queue (see below).

Return Value

```xml
<textmessage>
  <header>
    <JMSMessageID value="ID:Oc4jJMS.Message.dencmassive.4b6cd8a4:1198662edb3:-8000.27230" />
    <JMSExpiration value="0" />
    <JMSRedelivered value="false" />
    <JMSReplyTo value="null" />
    <JMSTimestamp value="1209411696521" />
    <JMSType value="" />
  </header>
  <properties>
    <string key="JMSXConsumerTXID" value="" />
    <int key="JMSXDeliveryCount" value="1" />
    <string key="JMSXProducerTXID" value="Oc4jJMS.Session.dencmassive.4b6cd8a4:1198662edb3:-8000.27228.0" />
    <long key="JMSXRcvTimestamp" value="1209411696521" />
    <string key="JMSXUserID" value="" />
    <string key="JMS_OC4J_Type" value="textmessage" />
  </properties>
  <textbody>
    <string value="<?xml version = '1.0' encoding = 'UTF-8'?>
    <jdeResponse pwd="" token="" role=""ALL" type="realTimeEvent" category="RTE" user="JDE" session="c71c90" environment="JDV812" on><version>ZJDE0001</version><sessionID>c71c90</sessionID><environment>JDV812</environment><host>DENCBELFORD</host><sequenceID>801</sequenceID><date>04282008</date><time>1</time><Info></header><body elementCount="1"><detail_D0100085A date="04282008" name="AddressBookMasterRealTimeWrapper" essType3/><cAddressType2/><szCategoryCodeAddressBook2/><szTaxId/><szEdiTranslationFormat/><szTypeTransaction/><cActionMessageControl/><szPhoneNumber1/><szCategoryCodeAddressBk29/><mnEdiDetailLinesProcess>0</mnEdiDetailLinesProcess><szCategoryCodeAddressBk28/><szCategoryCodeAddressBk27/><szGlBankAccount/><szCategoryCodeAddressBk24/><szCa ... dressBk23/><szCategoryCodeAddressBk22/><szTaxId2/><mnAddressNumber>55116</mnAddressNumber><mnTimeOfDay>0</mnTimeOfDay><szTon/><cAddressTypeEmployee/><szCreditMessage/><szCountry/><szUserId/><szReportCodeAddBook020/><jdDateBeginningdBook019/><szWorkStationId/><mnTimeScheduledIn>0</mnTimeScheduledIn><szReportCodeAddBook018/><szReportCodeAddBook011/><szReportCodeAddBook010/><mnAddressNumber6th>0</mnAddressNumber6th><szContactTilleNumber4th>0</mnAddressNumber4th><szTradingPartnerId/><cSubledgerInactiveCode/><jdUserReservedDate/><szReportCodeAddBook005/><szUserReservedCode/><szReportCodeAddBook004/><szAlternateAddressKey/><szReportCodeAddBook003/><mnEdiLineNumber>0</mnEdiLineNumber><szPhoneAreaCode2/><szPhoneAreaCode1/><szEdiBatchNumber11/><szMailingName><mnParentNumber>0</mnParentNumber><szUserReservedReference/><mnAddressNumber3rd>0</mnAddressNumber3rd><mnUserReservedAmount>0</mnUserReservedAmount><cEdiSuccessfulLastUpdated></detail_D0100085A></body></event></jdeResponse>" />
  </textbody>
</textmessage>
```
1 messages processed
Step 4: Import the “RIForOutbound.zip” to JDev.

1. Open JDev and import “testESB.jpr” from the extracted “RIForOutbound.zip” folder. I named the application “ESB_BSSV_RIOutbound”.

2. Open “testESB.esb” file
3. click on the icon on the “JMS_ReadF...” (JMS_ReadFromESBQueue). You'll see following window:
4. Click on the “Configure tech adapter service wsdl” icon right next to the “WSDL File: JMS_ReadFromESBQueue.wsdl”

5. The Adapter Configuration Wizard will start Click on Next twice and take the defaults.

![Adapter Configuration Wizard - Step 2 of 2: JMS Provider](image)

7. Click New to create a connection to your Transaction Server queue: Create Application Connection Wizard will show up (see below)
8. Type your connection name and select **Oracle Application Server 10g 10.1.3** for your Connection Type:

9. Enter your username: and Password: for your transaction server OAS and check “Deploy Password”.
6. Select "Single Instance"
   a. Host Name: your TS OAS name
   b. OPMN Port: See Step2 #3
   c. OC4J Instance Name: See Step2 #3
7. Test the connection and make you get “Success!!” message
8. Now, you should be back in the STEP 4 of the Adapter configuration Wizard page.

9. Enter your JMS Queue info:
   Destination Name (Queue):  jms/ESBS subscriberQueue (You can browse to it)
   JMS Connection Factory:  jms/QueueConnectionFactory
   Message Body Type:  TextMessage
   Message Selector:
   JNDI Name:  eis/Jms/RemoteQueueConnectionFactory
10. Open oc4j-ra.xml in <SOAinstall>/j2ee/home/application-deployment/default/jmsadater
11. Add below entry at the end of the file:

```xml
<connector-factory location="eis/Jms/RemoteQueueConnectionFactory" connector-name="Jms Adapter">
   <config-property name="connectionFactoryLocation" value="jms/QueueConnectionFactory" />
   <config-property name="factoryProperties" value="java.naming.factory.initial=oracle.j2ee.rmi.RMIInitialContextFactory;java.naming.provider.url=ormi://demnmassive.mlab.jdedwards.com:12404;java.naming.security.principal=oc4jadmin;java.naming.security.credentials=oc4jadmin" />
   <config-property name="acknowledgeMode" value="AUTO_ACKNOWLEDGE" />
   <config-property name="isTopic" value="false" />
   <config-property name="isTransacted" value="false" />
   <config-property name="username" value="oc4jadmin" />
   <config-property name="password" value="oc4jadmin" />
   <connection-pooling use="none" />
   <security-config use="none" />
</connector-factory>
```

12. Save the file and restart SOA Suite

---

**Step 5: Register and Test to Sample on the ESB Server**

1. Open JDeveloper
2. Start SOA Suite
3. Create **Application Server Connection** and **Integration Server Connection** for your SOA Suite
4. Click on the **Connections Tab** and right mouse click on **Application Server**

5. Type the Connection Name: and Select "**Oracle Application Server 10g 10.1.3**" for the Connection Type:
6. In the next screen Type your Username: and Password: for SOA Suite. Check the “Deploy Password”.
7. Select Single Instance, Type SOA suite host name, OPMN port: and OC4J instance name:
12. Test the connection and Finish

13. Right mouse click “New Integration Server Connection” on “Integration Server” in the Connections tab.
14. Enter the connection Name:
15. Select your Application Server Connection and take defaults:
16. Test the connection and finish
17. Go back to the "Applications Navigation" tab and right mouse click on testESB project
18. Remember, you have a message in JMSQueue(ESB/ESBSubscriberQueue) in your OAS. Select “Register with ESB” and select your Integration Server connection.
19. Wait until you see the successful message

20. Go to C:\outputdir on SOA machine. You should be able to see the XML doc with today’s date.

21. Go to ESB console, and you can view your ESB process (click on the “Instance” icon and click on the instance on the left window.)
22. As you can see, the JMS adapter picked the message up and routed to RS_RTABOUT and wrote a file using the file adapter.

COMPLETE!!!!